What is claimed is:

1. A computer program product for improving scheduling of tasks, the computer program product embodied on one or more computer readable media and comprising:

computer-readable program code means for computing whether execution of a plurality of tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

computer-readable program code means for adding an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when the execution is computed to be feasible;

computer-readable program code means for iteratively repeating operation of the computer-readable program code means for computing and the computer-readable program code means for adding, until the execution is computed to be no longer feasible; and

computer-readable program code means for using the revised cost for each task as an upper limit on execution time for the task, after operation of the computer-readable program code means for iteratively repeating.

2. The computer program product according to Claim 1, wherein the additional amount of time is a fixed percentage of the associated cost for the task.

3. The computer program product according to Claim 1, wherein the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task.

4. The computer program product according to Claim 1, wherein on a first iteration of the computer-readable program code means for adding, the additional amount of time is a fixed percentage of the associated cost for the task and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

5. The computer program product according to Claim 1, wherein on a first iteration of the computer-readable program code means for adding, the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task, and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

6. The computer program product according to Claim 1, wherein the computer-readable program code means for using further comprises:

computer-readable program code means for determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, computer-readable program code means for allowing the particular task to run until reaching a minimum of (1) an amount of time remaining until the task's associated deadline or (2) the upper limit on execution time for the task.

7. A system for improving scheduling of tasks, comprising:

means for computing whether execution of a plurality of tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

means for adding an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when the execution is computed to be feasible; and

means for iteratively repeating operation of the means for computing and the means for adding, until the execution is computed to be no longer feasible.

8. The system according to Claim 7, further comprising means for using the revised cost for each task as an upper limit on execution time for the task, after operation of the means for iteratively repeating.

9. The system according to Claim 7, wherein the additional amount of time is a fixed percentage of the associated cost for the task.

10. The system according to Claim 7, wherein the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task.

11. The system according to Claim 7, wherein on a first iteration of the means for adding, the additional amount of time is a fixed percentage of the associated cost for the task and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

12. The system according to Claim 7, wherein on a first iteration of the means for adding, the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task, and wherein on other iterations, the additional

amount of time is a fixed percentage of the revised cost for the task.

13. The system according to Claim 7, wherein the means for using further comprises:

means for determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, means for allowing the particular task to run until reaching a minimum of (1) an amount of time remaining until the task's associated deadline or (2) the upper limit on execution time for the task.

14. A method for improving scheduling of tasks, comprising steps of:

computing whether execution of a plurality of tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

adding an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when the execution is computed to be feasible; and

iteratively repeating operation of the computing step and the adding step, until the execution is computed to be no longer feasible.

15. The method according to Claim 14, further comprising the step of using the revised cost for each task as an upper limit on execution time for the task, after operation of the step of iteratively repeating.

16. The method according to Claim 14, wherein the additional amount of time is a fixed percentage of the associated cost for the task.

17. The method according to Claim 14, wherein the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task.

18. The method according to Claim 14, wherein on a first iteration of the adding step, the additional amount of time is a fixed percentage of the associated cost for the task and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

19. The method according to Claim 14, wherein on a first iteration of the adding step, the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task, and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

20. The method according to Claim 14, wherein the using step further comprises the steps of:

determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, allowing the particular task to run until reaching a minimum of (1) an amount of time remaining until the task's associated deadline or (2) the upper limit on execution time for the task.